



MZA Associates Corporation

Matlab-ODBC Interface

**Robert W. Praus, II
Stephen J. Verzi**

April 20, 1999

**MZA Associates Corporation
2021 Girard SE
Suite 150
Albuquerque, NM 87106**

**Voice: (505)245-9970
Fax: (505)245-9971**

praus@mza.com



Overview

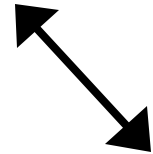
- **MZA has implemented a library of Matlab commands which allow users to interact with industry-standard databases (RDBMS).**
- **The software is very portable because it based on the Microsoft ODBC standard. It runs on Windows and Linux systems and supports all ODBC databases.**
 - All major databases including, Access, SQL Server, Oracle, Ingres, and Empress support the ODBC standard.
 - Part of the ODBC standard is SQL; procedures and mechanisms intended to provide SQL query capabilities can be implemented through this technique.
- **The user can load data from a database into Matlab or store data from Matlab into the database. He is provided a number of mechanisms for sorting, searching, and otherwise querying the database.**
- **By mapping database tables to Matlab structures with which Matlab users are more familiar, we reduce the adjustment that Matlab users must make when working with data from the database.**
- **ODBC connections to the database can be made by authorized users from anywhere on the internet.**
 - LAN connections will generally be fast and well-suited for large data retrieval.
 - WAN connections will generally be slow and well-suited only for looking at low-volume summary data.

Database Tables are Mapped to Matlab Structures

» % The structure in Matlab

» whos t

```
Name      Size      Bytes Class
t         1x10      12384 struct array
```



The Table in Access

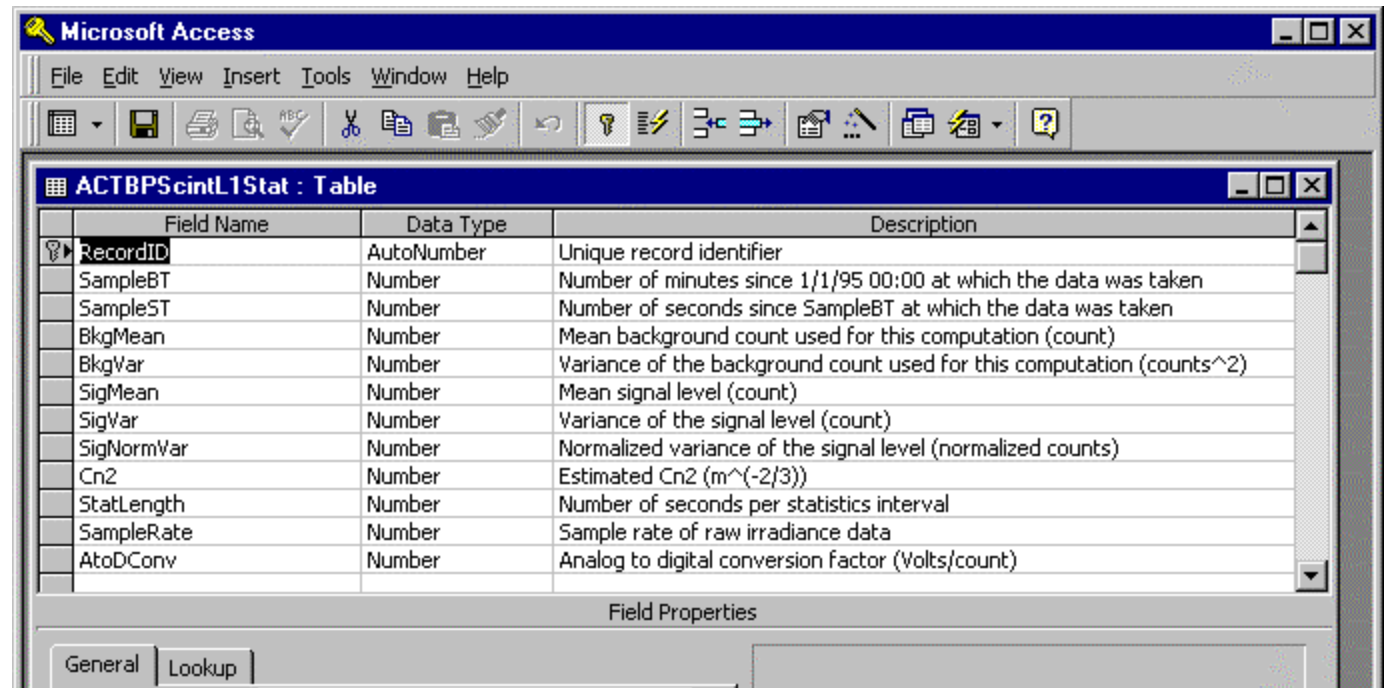
Grand total is 240 elements using 12384 bytes

» t

t =

1x10 struct array with fields:

```
RecordID
SampleBT
SampleST
BkgMean
BkgVar
SigMean
SigVar
SigNormVar
Cn2
StatLength
SampleRate
AtoDConv
```



Field Name	Data Type	Description
RecordID	AutoNumber	Unique record identifier
SampleBT	Number	Number of minutes since 1/1/95 00:00 at which the data was taken
SampleST	Number	Number of seconds since SampleBT at which the data was taken
BkgMean	Number	Mean background count used for this computation (count)
BkgVar	Number	Variance of the background count used for this computation (counts^2)
SigMean	Number	Mean signal level (count)
SigVar	Number	Variance of the signal level (count)
SigNormVar	Number	Normalized variance of the signal level (normalized counts)
Cn2	Number	Estimated Cn2 (m^(-2/3))
StatLength	Number	Number of seconds per statistics interval
SampleRate	Number	Sample rate of raw irradiance data
AtoDConv	Number	Analog to digital conversion factor (Volts/count)



Example Matlab Commands

- **Example Matlab commands:**

- » `[dbid,rc] = mdbopen('mmactdual','sverzi','xxxxxxx');`
- » `[tabs,rc] = mdbtables(dbid);`
- » `[cols,rc] = mdbcolumns(dbid,'ACTScintStatsL3');`
- » `[ts,rc] = mdbfetch(dbid,'ACTScintStatsL3','time,cn21,cn22',...
'time BETWEEN 19981217000000000000 AND 19981218000000000000','time');`
- » `[ts,rc] = mdbfetch(dbid,'ACTScintStatsL3','time,cn21,cn22',...
'time BETWEEN 19981201000000000000 AND 19990101000000000000','time');`
- » `[rc,sql] = mdbcreate(dbid,'MyScintStats30',myts,types);`
- » `[rc,sql] = mdbinsert(dbid,'MyScintStats30',myts);`
- » `[rc] = mdbclose(dbid);`



MATLAB-ODBC Interface

- **The MATLAB-ODBC interface (MODBC) consists of 8 top-level functions:**
 - **mdbopen** - open a database connection.
 - **mbclose** - close a database connection.
 - **mbcreate** - create a database table.
 - **mbdrop** - drop a database table.
 - **mbinsert** - insert a MATLAB structure into a database table.
 - **mbfetch** - fetch a database table into a MATLAB structure.
 - **mbtables** - get names and types of tables in a database.
 - **mbcolumns** - get column information for a table in a database.

- **MODBC is a toolset implemented for accessing data both interactively and in batch programs. It is not a database access GUI.**

Connect to the Dual Source Database on madmax.plk.af.mil.

```
» [dbid,rc] = mdbopen('mmactdual','sverzi','xxxxxxx')
```

dbid =

Columns 1 through 5

```
[1002] [1x4 uint8] {1x3 cell} 'mmactdual' 'sverzi'
```

Columns 6 through 8

```
'xxxxxxx' [1] {}
```

rc =

0

Authorized users will be assigned their own usernames and passwords.



MZA Associates Corporation

mdbtables

The Dual Source database has 17 tables.

```
» [tabs,rc] = mdbtables(dbid)
```

```
tabs =
```

```
1x17 struct array with fields:
```

```
    Name
```

```
    Type
```

```
rc =
```

```
    0
```

```
» strvcat(tabs.Name)
```

```
ans =
```

```
ACTDualSourceLog
```

```
ACTScintBeam1StatsL2
```

```
ACTScintBeam2StatsL2
```

```
ACTScintProcStatsL4
```

```
ACTScintProcWindSpL4
```

```
ACTScintStatsL3
```

```
ACTScintWindSpL3
```

```
ACTTracker1DTIMGL3
```

```
ACTTracker1IMGL2
```

```
ACTTracker1IMGL3
```

```
ACTTracker1SCLL1
```

```
ACTTracker1SCLL3
```

```
ACTTracker2DTIMGL3
```

```
ACTTracker2IMGL2
```

```
ACTTracker2IMGL3
```

```
ACTTracker2SCLL1
```

```
ACTTracker2SCLL3
```



MZA Associates Corporation

mdbcolumns

The table ACTScintStatsL3 has time-stamped C_n^2 values for both beams as well as means and variances.

```
» [cols,rc] = mdbcolumns(dbid,'ACTScintStatsL3')
```

```
cols =
```

```
1x9 struct array with fields:
```

```
Name  
DataType  
TypeName  
Precision  
Length  
Scale  
Radix  
Nullable  
Remarks
```

```
rc =
```

```
0
```

```
» strvcat(cols.Name)
```

```
ans =
```

```
time  
i1mean  
i1var  
i2mean  
i2var  
i1nvar  
i2nvar  
cn21  
cn22
```



mdbfetch

For December 17, 1998, fetch the ACTScintStatsL3 table and plot the C_n^2 values for both beams.

```
» [ts,rc] = mdbfetch(dbid,'ACTScintStatsL3','time,cn21,cn22',...  
    'time BETWEEN 19981217000000000000 AND 19981218000000000000',...  
    'time')
```

1000 of 1452 rows fetched in 1 seconds

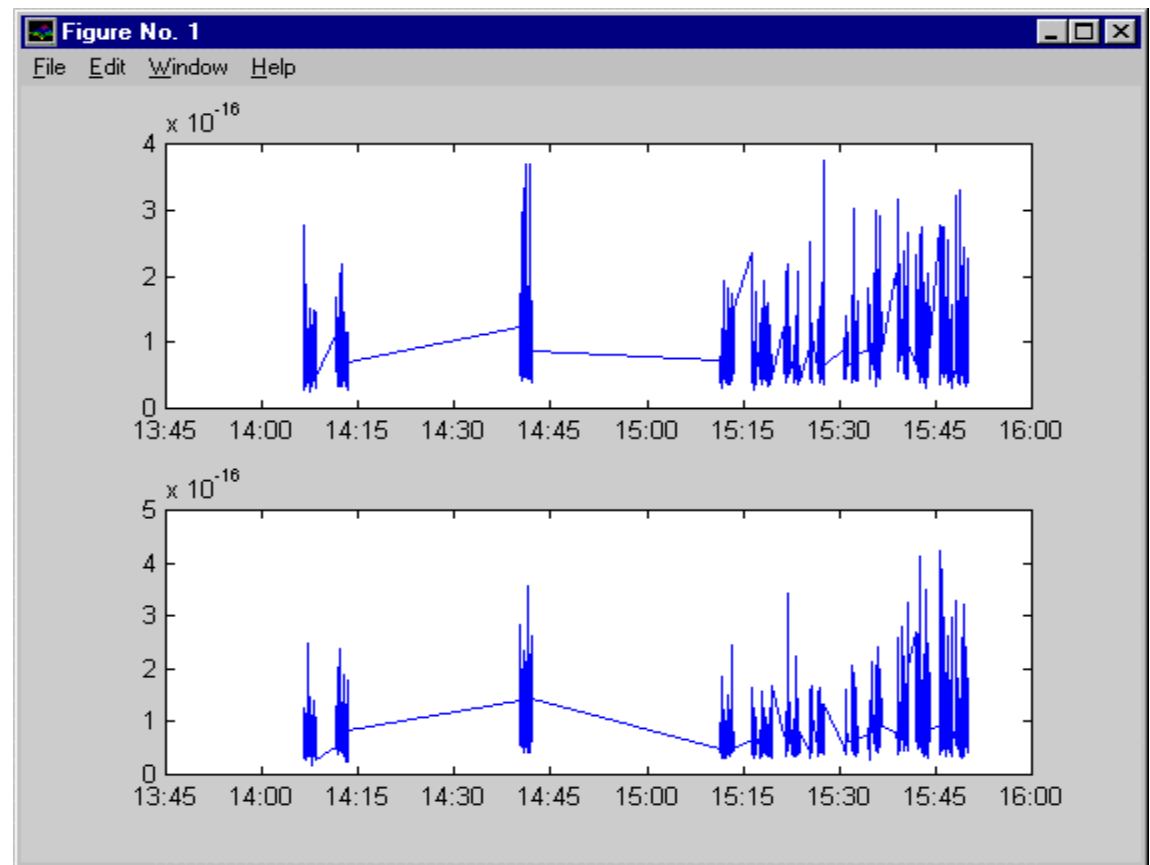
ts =

```
time: [1x1452 double]  
cn21: [1x1452 double]  
cn22: [1x1452 double]
```

rc =

0

```
» subplot(2,1,1),plot(ts.time,ts.cn21);  
» axes('','','','h:m');  
» subplot(2,1,2),plot(ts.time,ts.cn22);  
» axes('','','','h:m');
```





MZA Associates Corporation

mdbfetch

Fetch time-stamped Cn2 (1-second averages) values for the month of December 1998 and plot beam1 versus beam2.

```
» [ts,rc] = mdbfetch(dbid,'ACTScintStatsL3','time,cn21,cn22',...  
    'time BETWEEN 19981201000000000000 AND 19990101000000000000',...  
    'time')
```

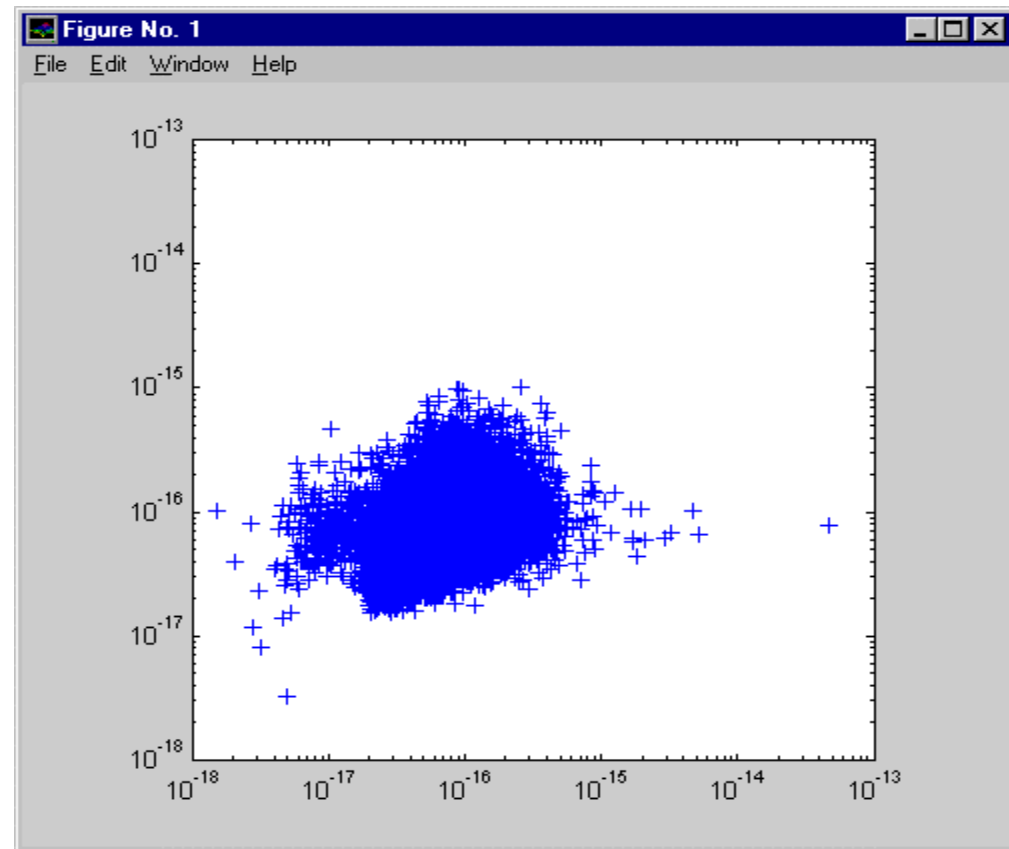
ts =

```
time: [1x36552 double]  
cn21: [1x36552 double]  
cn22: [1x36552 double]
```

rc =

0

```
» loglog(ts.cn21,ts.cn22,'+')
```

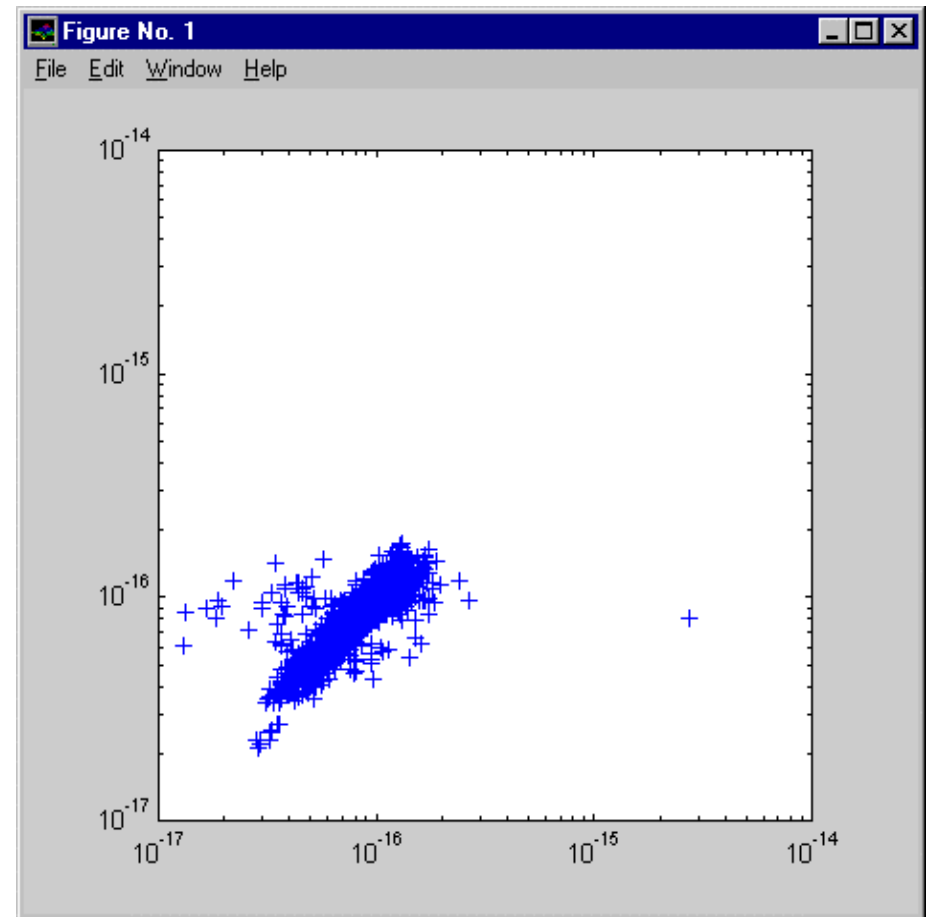




Simple Processing

Calculate 30-second average C_n^2 values using the 1-second averages and plot beam1 versus beam2.

```
» myts.time = mean(reshape(ts.time(1:36000),30,1200));  
» myts.cn21 = mean(reshape(ts.cn21(1:36000),30,1200));  
» myts.cn22 = mean(reshape(ts.cn22(1:36000),30,1200));  
» loglog(myts.cn21,myts.cn22,'+')
```



A user tries to create a table without appropriate privilege.

```
» [rc,sql] = mdbcreate(dbid,'MyScintStats30',myts,types)
```

```
Warning: mdbcreate: Error - executing CREATE statement.
```

```
> In d:\sverzi\modbc\mdbcreate.m at line 132
```

```
rc =
```

```
-1
```

```
sql =
```

```
CREATE MyScintStats30(time microtimestamp, cn21 float, cn22 float)
```



mdbcreate

A user with privilege creates the table MyScintStats30.

```
» [rc,sql] = mdbcreate(dbid,'MyScintStats30',myts,types)
```

```
rc =
```

```
0
```

```
sql =
```

```
CREATE MyScintStats30(time microtimestamp, cn21 float,  
cn22 float)
```

```
» [tabs] = mdbtables(dbid);  
» strvcat(tabs.Name)
```

```
ans =
```

```
ACTDualSourceLog  
ACTScintBeam1StatsL2  
ACTScintBeam2StatsL2  
ACTScintProcStatsL4  
ACTScintProcWindSpL4  
ACTScintStatsL3  
ACTScintWindSpL3  
ACTTracker1DTIMGL3  
ACTTracker1IMGL2  
ACTTracker1IMGL3  
ACTTracker1SCLL1  
ACTTracker1SCLL3  
ACTTracker2DTIMGL3  
ACTTracker2IMGL2  
ACTTracker2IMGL3  
ACTTracker2SCLL1  
ACTTracker2SCLL3  
MyScintStats30
```

Insert calculated statistics into MyScintStats30.

```
» [rc,sql] = mdbinsert(dbid,'MyScintStats30',myts)
1000 (of 1200) rows inserted in 10.615 seconds.
```

```
rc =
```

```
0
```

```
sql =
```

```
INSERT INTO MyScintStats30 VALUES(?,?,?)
```



mdbdrop

Drop MyScintStats30 from the Dual Source database.

```
» [rc,sql] = mdbdrop(dbid,'MyScintStats30')
```

```
rc =
```

```
0
```

```
sql =
```

```
DROP MyScintStats30
```

```
» [tabs] = mdbtables(dbid);  
» strvcat(tabs.Name)
```

```
ans =
```

```
ACTDualSourceLog  
ACTScintBeam1StatsL2  
ACTScintBeam2StatsL2  
ACTScintProcStatsL4  
ACTScintProcWindSpL4  
ACTScintStatsL3  
ACTScintWindSpL3  
ACTTracker1DTIMGL3  
ACTTracker1IMGL2  
ACTTracker1IMGL3  
ACTTracker1SCLL1  
ACTTracker1SCLL3  
ACTTracker2DTIMGL3  
ACTTracker2IMGL2  
ACTTracker2IMGL3  
ACTTracker2SCLL1  
ACTTracker2SCLL3
```

WARNING: a dropped table and all of its contents are removed from the database permanently!



mdbdrop

A deviant user attempts (unsuccessfully) to drop ACTScintStatsL3 from the Dual Source database.

```
» [rc,sql] = mdbdrop(dbid,'ACTScintStatsL3')
```

Warning: mdbdrop: Error - executing DROP statement.

```
> In d:\sverzi\modbc\mdbdrop.m at line 49
```

```
rc =
```

```
-1
```

```
sql =
```

```
DROP ACTScintStatsL3
```

```
» [tabs] = mdbtables(dbid);  
» strvcat(tabs.Name)
```

```
ans =
```

```
ACTDualSourceLog  
ACTScintBeam1StatsL2  
ACTScintBeam2StatsL2  
ACTScintProcStatsL4  
ACTScintProcWindSpL4  
ACTScintStatsL3  
ACTScintWindSpL3  
ACTTracker1DTIMGL3  
ACTTracker1IMGL2  
ACTTracker1IMGL3  
ACTTracker1SCLL1  
ACTTracker1SCLL3  
ACTTracker2DTIMGL3  
ACTTracker2IMGL2  
ACTTracker2IMGL3  
ACTTracker2SCLL1  
ACTTracker2SCLL3
```

mdbclose

End a database session by closing the database connection.

```
» [rc] = mdbclose(dbid)
```

```
rc =
```

```
0
```